# The Single Logon Application: An Enabler for Access to Disparate Systems in a Patient-Focused Care Environment

Spencer L. SooHoo[1], Ph.D., Fred Aabedi[1], Ralph C. Wagenet[1], Mark Dorst[2], and Maggie Stempson[1], RN, MSN
[1]Cedars-Sinai Health Systems Los Angeles, California
[2]Walker Richer and Quinn Seattle, Washington

*The presence of separate, independent systems for patient results is a common problem in many institutions. The Single Logon Application (SLA) was developed to mitigate the problems presented by these disparate systems, and its use was a key component in a Patient-Focused Care (PFC) implementation.*

## INTRODUCTION

As health care institutions restructure to survive the transition from a fee-for-service to a capitated environment, the challenge is to reduce human resources and some traditional services while maintaining a high level of care. At Cedars-Sinai Medical Center (CSMC), a major element of the restructuring has been the Patient-Focused Care (PFC) concept. This entailed a complete redesign of work flow and job responsibilities, deployment of equipment, supplies, and support services directly to patient care areas when economically sensible, and giving employees in these areas both the authority and accountability for managing patient care resources. Towards this end, the "center" for patient care was migrated from the traditional central nursing station to smaller stations with a team that was responsible for most of the patient care and clerical and administrative functions.

From an information technology viewpoint, this reconfiguration required a redesign of our current access to the hospital's computer systems. Associated with the central nursing station concept were "dumb" terminals that were used primarily by clerical staff for Admission, Discharge, Transfer (ADT) functions, order-entry, and results inquiry. Although all support and licensed nursing personnel as well as many physicians had been trained in the use of the terminals for order entry and results inquiry, these systems were rarely used by them and instead, there was a heavy reliance on clerical staff to perform these functions. Among the reasons were (a) there were too few terminals that were often poorly sited with respect to the patient rooms (b) the terminals were connected to disparate systems with their own "look and feel" and separate logon procedures. The latter results in problems with remembering username and password combinations and can discourage most of the non-clerical users.

The use of multiple, often independent computer systems is common in many health care institutions (1-4). For example at CSMC, the ADT, pharmacy, and radiology tracking systems are on an International Business Machines (IBM) mainframe computer while, the laboratory system, drug interaction, and various specialty data bases (such as Cardiovascular Surgery tracking) are on a Digital Equipment Corporation (DEC) VAX cluster. And the new pharmacy and radiology systems that are scheduled for installation will be on hardware running a UNIX operating system.

In 1991, Information Systems Services (ISS) at CSMC started a project to address the problems associated with having disparate systems for patient care. The outcome was the Integrated Viewing System (IVS) which essentially embodies the concept of a data warehouse or central clinical repository and is similar in concept to efforts at other institutions (2,4-7). As illustrated in Figure 1, IVS is a relational database that gets results from numerous "feeder systems" like the clinical and pulmonary systems. IVS can also relay the data to other specialized systems like our Surgical Intensive Care Unit (SICU) monitoring system. The primary links between the various systems are either Health Level Seven (HL-7) or file transfer; the latter is used primarily for systems that cannot support HL-7 and which supply mostly textual data like transcribed reports. A notable exception is the ADT system, where HL-7 would be too costly, so what amounts to terminal emulation was used to capture ADT transactions.

When fully implemented, IVS will reduce the number of systems that had to be used to do results inquiry to two. The roadblocks to full implementation are more procedural rather than technical in nature, but the process is well underway with the major items (clinical laboratory, blood gases, pulse oxime-

0195-4210/95/$5.00 © 1995 AMIA, Inc.

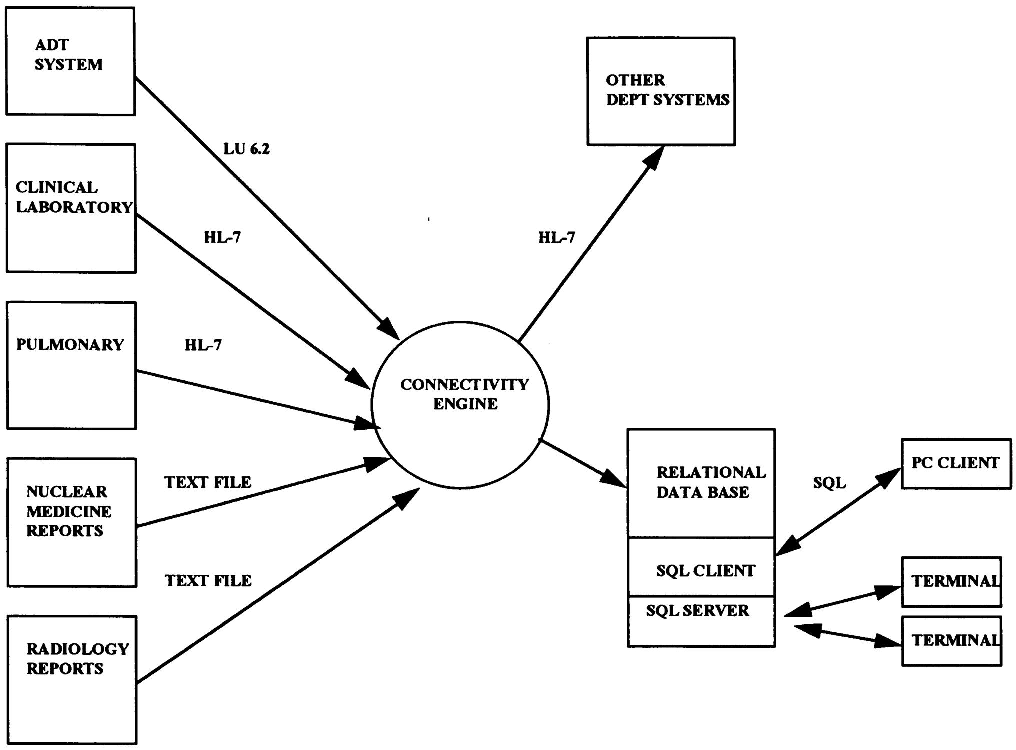


Figure 1. Overview of IVS.

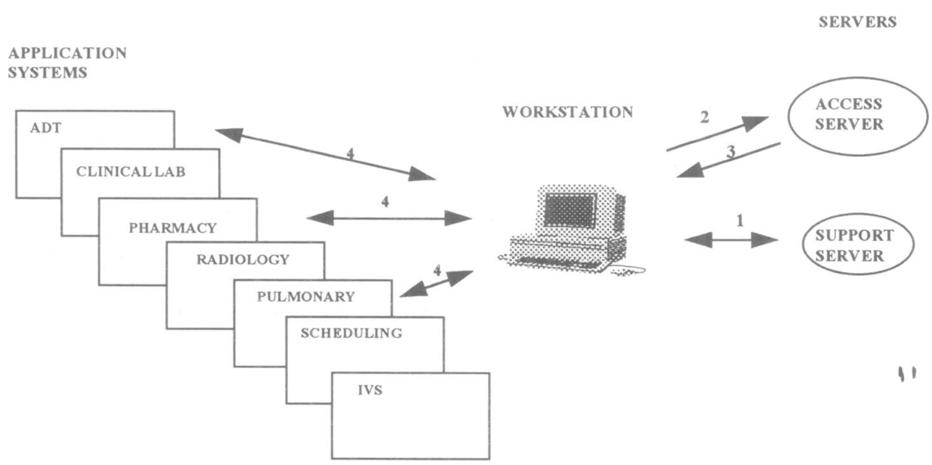


Figure 2. SLA overview showing, applications, access server, support server, and workstation.

try monitoring, pulmonary function reports, transcribed anatomic pathology results, nuclear medicine and radiology reports, a cardiovascular surgery database, and many elements of the ADT records) available through IVS. Users access the ADT system only to peform the actual admission, discharge, or transfer of a patient; viewing the patient demographics and insurance information portion of the ADT record can be done via IVS.

IVS also provides a "pass through" function so that other systems such as a Medline data base and ambulatory patient scheduling will be accessible through a "single logon." The main drawback of this approach was that it depended on VAX-based host system to do the "pass through" function, and did not take advantage of the processing power and software that are now available on desktop systems like the Personal Computer (PC).

The initial deployment of IVS was on existing "dumb" terminals, and it was envisioned that the next set of enhancements would involve workstations based on the PC (IBM compatible) so that we could take advantage of the graphical user interface (GUI) and lower software development costs on those platforms and position the Medical Center to take advantage of client-server technology as applications became available.

The decision to implement PFC escalated the planning effort. We recognized that we would have to maintain our existing mainframe and minicomputer-based systems for the immediate future but we also wanted as much as possible, to give the user the illusion that there was a single system that they were using without a protracted software development effort, since we only had about 6 months to come up with a solution.

## DESIGN REQUIREMENTS

1. Accommodate existing and future "dumb" terminal applications that run on the mainframe and minicomputers.
2. Accommodate newer client-server software such as a new Sybase-based radiology scheduling system,
3. If possible, avoid making code changes to existing applications.
4. Give users a single logon name and password across all applications, regardless of the operating system or hardware platform.
5. Provide automatic password synchronization among all applications.
6. Provide user-specific profiles so that the same workstation can be used by different clinical specialists (such as nursing and pharmacy).
7. Allow transfer of key information such as the medical record number and patient name between applications.
8. Use Microsoft Windows, since this has become a de-facto standard at this institution.
9. Provide "remote control" so that help desk staff can assist users from a central location.
10. Provide automated, centralized software distribution since there will be over 200 workstations spread across a large campus.
11. Minimize single points of failure so that workstations will still function without reliance on a single major sub-system.
12. Minimize support problems due to configuration changes or network problems.

## BASIC APPROACH

Our strategy is conceptually similar to other implementations (8,9). As shown in Figure 2, the key components consist of an Access Server, the PC workstation loaded with the SLA software, the various host and application servers, and a Novell file server.

The client hardware consists of a PC (Intel Pentium with 16 MB memory, 540 MB hard disk) with Microsoft Windows 3.1 or Windows for Workgroups 3.11 As previously mentioned, the Access Server is an existing VAX cluster, which was selected since it was configured for high availability with automatic failover and redundant (mirrored) disk drives.

The Access Server code is written in DCL, the command language that is provided on DEC's VMS operating system. It makes use of the existing username and password file, so from an operating system viewpoint, Access Server requests are treated like normal interactive logon sessions.

The workstation code is written in Visual Basic. A schematic diagram of the client configuration is shown in Figure 3. The code takes advantage of the scripting features of terminal emulators from Walker Richer and Quinn (WRQ). We used Reflections 3270 for IBM 3270 and Reflections 2 for DEC VT-

| VISUAL BASIC SHELL | |
|---|---|
| IBM TERM EMULATOR | VT TERM EMULATOR |
| WINDOWS 3.1 | |
| SUPPORT TSR TCP/IP AND IPX DRIVERS | |
| DOS 6.22 | |

Figure 3. SLA workstation configuration.

200 terminals in conjunction with a combination of Windows Dynamic Data Exchange (DDE) and Application Program Interface (API) calls to pass data between the sessions and the Visual Basic SLA program.

Other elements of the PC configuration are the support software (TSR support and IPX drivers) which allow for the remote control and automated software distribution and the network connection to the application servers and host systems which are via the TCP/IP protocol stack. This connection is represented by arrow 1 in Figure 2.

The SLA program gets a username and password from the user and passes it to the Reflections 2 emulator which initiates a standard logon session on the VAX (arrow 2 in Figure 2). Upon a successful logon, the Access server passes tokens back to the PC (arrow 3 in Figure 2). The tokens represent the system, application, and username and encrypted password of the application that the user has access to.

When the client software receives the token, it is stored in memory and used to create icons representing each application. As the user clicks on one of these icons, the SLA program initiates a logon session for that application using the username and password that form the token (arrow 4 in Figure 2). When the session is established, it appears as the primary window on the workstation and a green border is drawn around the icon to indicate that it is the active session. If a second icon is selected, a second logon session is established and becomes the active session. The previous active session is hidden and the border around it is changed to yellow to indicated that it is active but hidden. If a hidden session is timed out, the border turns gray. Up to eight different applications can be active under the current version of SLA.

Password synchronization is carried out by having the SLA client invoke the change password function for the individual application the user has access to. This assumes that when the user is first setup for SLA, the passwords on these systems is set to a standard default. And, the password lifetime on the SLA Access Server is shorter than that on the other host systems so that password changes are required by the SLA server first.

A "cut and paste" function allows the user to pass information such as a medical record number or patient name between applications.

The remote software distribution and remote control aspects were implemented on a Novell File Server. This allowed us to take advantage of the Novell system announcement and broadcast features for posting system downtime notices, etc.. The client SLA software is configured so that if the file server is not available, the application can continue running and the only impact will be that support staff will be unable to provide remote support until the file server is repaired.

The current host systems consist of our ADT system which runs on an IBM mainframe, a Pharmacy system that runs on a UNIX platform, and two VAX/VMS applications, IVS, our clinical data repository, and Micromedix, a drug interaction database.

Since we wanted to build the system to be as robust as possible, we elected to put it on its own Ethernet segment and to strictly control the software configuration. Towards this end, we put in a floppy disk lock and disabled the Windows Task Manager so that the user has no choice but to use the SLA and does not have access to the normal Windows applications or tools. This insures that the application is as stable as we can make it from both a networking and software configuration viewpoint.

## RESULTS

The overall response to the SLA has been very positive. The number and siting of the PC workstations makes them readily accessible, and by eliminating multiple accounts, the user is more likely to use the system.

Some of the problems we have encountered are:

1. There are some problems with the different systems having their unique "look and feel", but for the PFC areas, this is minimal since the users deal mostly with two different systems, one for ADT and one for IVS.
2. Microsoft Windows is not as robust an environment as we would like. However, we have developed workarounds, and the current SLA software is stable.
3. Password synchronization is a difficult problem to deal with if the implementation phase is protracted., since users can go to "dumb" terminals and change their password for one or more applications, causing the Access Server to lose password synchronization.
4. We find it difficult to keep up with the demand to add more applications to the SLA. The testing effort to insure compatibility across all applications is very demanding, and lack of good multitasking code in Windows makes this task difficult.

## FUTURE ENHANCEMENTS

The primary future enhancements include the implementation of a Hypertext Markup Language (HTML) viewer, support for local Windows applications, and electronic mail. The hypertext viewer is very important since we have made a strategic decision that we will use hypertext as a uniform means for distribution of information throughout the Medical Center. We plan to put policy and procedure manuals, clinical care guidelines, etc. on a centrally managed hypertext server.

We are also studying the feasibility of migrating the SLA to Windows NT since we feel that this is a more robust environment, and we can take advantage of some of the integrated management tools like remote software distribution and configuration management that NT provides.

## REFERENCES


1. Pryor DB, Califf RM, Harrell F Jr. et al. Clinical data bases. Accomplishments and unrealized potential. Med Care 1985:23:623-47
2. Newman TB, Brown A, and Easteling MJ. Obstacles and Appoaches to Clinical Database Research: Experience at the University of California, San Francisco. Proceedings of the 18th Annual Symposium on Computer Applications in Medical Care. 18:568-572.
3. Bleich, HL and WV Slack. Designing a hospital information system: A comparision of interfaced and integrated systems. MD Computing, 1992; 9:293-296.
4. Clayton, PD, RV Sideli, and S Sengupta. Open architecture and integrated information at Columbia-Presbyterian Medical Center. MD Computing, 1992; 9:297-303.
5. Hammond, JE, RG Berger, TS Carey, SM Fakhry, R Rutledge, JP Kichak, TJ Cleveland, MJ Dempsey, NM Tsongalis, CF Ayscue Report on the Clinical Workstation and Clinical Data Repository Utilization at UNC Hospitals. Proceedings of the 18th Annual Symposium on Computer Applications in Medical Care. 18:276-277
6. Johnson, SB, G Hripcsak, J Chen, P Clayton. Accessing the Columbia Clinical Repository. Proceedings of the 18th Annual Symposium on Computer Applications in Medical Care. 18:281-285.
7. Hammond, JE, RG Berger, TS Carey, SM Fakhry, R Rutledge, JP Kichak, TJ Cleveland, MJ Dempsey, NM Tsongalis, CF Ayscue Progress Report on the Clinical Workstation and Clinical Data Repository Utilization at UNC Hospitals. Proceedings of the 17th Annual Symposium on Computer Applications in Medical Care. 17:243-247.
8. Barrows, RC, B Allen, and D Fink. An X Window System for Statlab Results Reporting. Proceedings of the 17th Annual Symposium on Computer Applications in Medical Care. 17:331-335.
9. Silva, JS, AJ Zawilski, J O'Brian, N Gunby, J Siegel, M Lauteren, R Halley. The physician workstation: An Intelligent "Front End" to a Hospital Information System. Proceedings of the 14th Annual Symposium on Computer Applications in Medical Care.14:764-768.